AD-A100 460 TEXAS UNIV AT AUSTIN CENTER FOR CYBERNETIC STUDIES F/G 12/1
AN ALGORITHM TO SELECT THE BEST SUBSET FOR A LEAST ABSOLUTE VAL--ETC(U)
MAY 81 R D ARMSTRONG, M T KUNG N00014-81-C-0236
UNCLASSIFIED CCS-396 NL

LEVEL II

12

CENTER FOR CYBERNETIC STUDIES

The University of Texas
Austin, Texas 78712

(12)

(14) CCS-396

AN ALGORITHM TO SELECT THE BEST SUBSET FOR A LEAST ABSOLUTE VALUE REGRESSION PROBLEM

by

(10) R. D. Armstrong*
M. T. Kung**

(12) 27

(11) May 1981

*Associate Professor, Operations Research and Statistics, Graduate School of Business, University of Texas, Austin, Texas

**Assistant Professor, Production and Management Science, Faculty of Business, McMaster University, Hamilton, Ontario

(15) N00014-81-C-0236, N00014-75-C-0569

This research was supported by ONR Contracts N00014-81-C-0236 and N00014-75-C-0569, with the Center for Cybernetic Studies, The University of Texas at Austin. Reproduction in whole or in part is permitted for any purpose of the United States Government.

CENTER FOR CYBERNETIC STUDIES

A. Charnes, Director
Business-Economics Building, 203E
The University of Texas at Austin
Austin, Texas 78712
(512) 471-1821

406177

ABSTRACT

This paper considers the problem of obtaining the best subset of regressors under a least absolute value criterion. The model is the classic linear regression model with m explanatory variables and a dependent variable. The importance of the explanatory variables is measured by obtaining the minimum sum of absolute deviations when only k of the m explanatory variables are included in the model. An algorithm is presented to obtain the "best" subset of size k, k = 1,...,m.

Several algorithms to solve the best subset problem are available when the criterion for evaluation is least squares. However, recently statisticians have become increasingly aware of the limitations of least squares and have popularized "robust-resistant" estimation techniques. Least absolute values is such a technique. Special purpose computer codes which utilize the simplex algorithm of linear programming are used to solve the least absolute value regression problem.

This paper incorporates two of these specialized codes within a branch-and-bound algorithm to solve the best subset problem. The advantages and disadvantages of the two codes, one primal and one dual, will be discussed. Also, a detailed description of the branch-and-bound implementation and the results of computational testing will be given.

KEY WORDS

Best Subset Regression Problem
Least Absolute Value
Linear Programming
Branch-and-Bound Algorithm

AN ALGORITHM TO SELECT THE BEST SUBSET FOR A LEAST ABSOLUTE VALUE REGRESSION PROBLEM

1. Introduction

In the experimental design models using regression analysis, it is common practice to examine different mathematical model formulations. As stated by Draper and Smith [6], there are two opposing criteria for establishing a useful and efficient regression equation. They are as follows:

(i) To make the equation useful for predictive purposes, it is advantageous to include as many independent variables as possible so that reliable fitted values can be determined.

(ii) Because of the costs, in terms of money, labor, and time, involved in obtaining information about a large number of independent variables and subsequently monitoring them, it is preferable to formulate the regression equation with as few independent variables as possible.

The compromise between these extremes leads to the notion of selecting the best subset of independent variables, or the best regression equation.

The problem of selecting variables has been well discussed in the literature in relation to the minimization of the sum of squared error criterion. The well-known procedures include various methods of selection from all possible regressions, backward elimination, forward selection and the stagewise regression procedure.

In recent years, the least absolute value criterion has been recommended in certain cases as an alternative to the least squares (see [14]). For investment models (see [12]) and for economic models with errors having non-finite variance (see [7]), the least absolute value criterion provides a more robust estimator than the least squares criterion. Kennedy and Gentle [9] provide an excellent review of solution procedures. It is the purpose of this paper to inspect algorithmic procedures to obtain the best subset of independent variables for a linear regression model under a least absolute value criterion.

2. The Best Subset Regression Problem

To estimate parameters for a multivariate linear regression model, the problem for the least absolute value criterion is stated as follows.

Given a set of n observational measurements $(y_i, x_{i1}, x_{i2},\ldots, x_{im})$, $i = 1,2,\ldots,n$, determine the values of the parameters, β_j, $j \in J$, which will

$$(1)\quad \underset{\beta}{\text{minimize}} \sum_{i=1}^{n} \left| y_i - \sum_{j \in J} x_{ij}\beta_j \right|$$

where $J \subseteq \{1,2,\ldots,m\}$ is the index set of the parameters in the model.

Charnes, Cooper and Ferguson [4] have shown that the optimal values of the parameters in (1) can be obtained via a linear programming formulation. Employing their result here, problem (1) is equivalent to:

$$(2)\quad \text{Minimize} \sum_{i=1}^{n} (P_i + N_i)$$

subject to

$$\sum_{j \in J} x_{ij}\beta_j + P_i - N_i = y_i, \quad i=1,2,\ldots,n,$$

$$P_i \geq 0, \quad N_i \geq 0, \quad i = 1,2,\ldots,n,$$

where P_i and N_i are, respectively, the positive and negative deviation associated with the i-th observation.

The "best" subset of a given number of independent variables, k, where $k \leq m$, is one which yields the minimum objective value of all possible subsets of k variables from among the set of m variables under consideration.

There are $m!/[k!(m-k)!]$ ways of choosing k regressors from a set of m regressors. In theory, each combination can be enumerated and solutions obtained and analyzed to determine the best subset of k regressors. However, total enumeration is generally not computationally efficient. Recently, implicit enumeration procedures have been developed to find the best subset of k regressors without examining all possible subsets. Beale, Kendall, and Mann [3], LaMotte and Hocking [10], and Furnival and Wilson [8] developed procedures to achieve this purpose under the least squares criterion. Two algorithms using the least absolute value criterion have been reported. Roodman [15] gives a partial enumerative search procedure using an upper bounding simplex algorithm to solve the dual of (2) and binary decision variables to specify the subset of regressors being assigned in the regression problem at each stage. Narula and Wellington [13] use an enumerative procedure that employs both a primal and a dual simplex algorithm along with a pre-optimality test which may

indicate suboptimality before a simplex iteration.

The algorithm proposed in this paper uses a branch-and-bound technique to find the best subset regression. This algorithm has features which are not present in the Roodman and Narula-Wellington algorithms. It uses a binary tree for enumerative purposes. Also, a rule for selecting a parameter to be restricted is introduced. Two linear programming algorithms are implemented separately to solve the least absolute value regression problems. The first one is a dual approach developed by Armstrong and Kung [2] while the other one is a primal method by Armstrong, Frome, and Kung [1]. Both approaches utilize information obtained from the solution of previous regressions to provide an advanced starting solution for the least absolute value regression problem currently considered. Like the Roodman and Narula-Wellington algorithms, bounding tests are also considered in this best subset regression algorithm.

The algorithm is presented in three parts. First, the branch-and-bound framework is outlined. Next, the special characteristics of this algorithm when using a dual linear programming method are given. In the last part, the implementation of the primal simplex approach within the branch-and-bound algorithm is described.

3. The Branch-and-Bound Framework

Enumerative algorithms are usually easier to understand if they are related to a tree composed of nodes and branches. Here, a node corresponds to a least absolute value regression problem

containing a specified set of parameters. This subproblem may be stated and solved as a linear programming problem of the form given by (2). The initial subproblem contains all the regression parameters in the model. After a subproblem is solved, the associated node is either fathomed or two descendants are created from it. Fathoming occurs when it can be ascertained that no regression problems of interest exist in any descendants of the node. If two descendants are created, they differ in the states of the parameters, where a parameter is forced out of the model in one node and the same parameter is required to be included in the model in the other node. The criterion for selecting the parameter to be restricted is the following.

From a list of free parameters (that is, the parameters that are not fixed to be in or out of the model), it is advantageous to select for restriction the parameter, which, when removed from the model, gives the least change in the optimal objective value. Thus, the best solutions should be examined earlier in the algorithm. Other subproblems which yield inferior solutions need not be solved. An intuitively appealing rule is to select the free parameter whose removal from the model will result in the smallest objective change during the first dual simplex iteration of the subsequent problem.

$$(3)\quad \beta_r = \min_{j \in F} \{\text{first iteration objective change when } \beta_j = 0\}$$

where F is the index set of the free parameters. This rule is based on the supposition that the first iteration reflects the overall objective change.

Once a parameter, β_r, is chosen to be restricted, one of the two descendant nodes deletes β_r from the problem, while the other node forces β_r to be included in the problem. Once a specification is established in a parent problem, it must also be satisfied in every descendant that follows. The restriction of parameters and the creation of more branches and nodes continue until there are no free parameters.

A solution tree for a four parameter problem is used for illustrative purposes. The complete structure of the tree is shown in Figure 1. It is assumed that the parameters to be fixed based on (3) are in the following hierarchical fashion: β_1, β_3, β_4, and β_2. The nodes in Figure 1 show the parameters included in the model. At each node, the right hand branch indicates the deletion of a specified parameter, and this parameter remains in the model on the left branch.

As seen in Figure 1, more than one node corresponds to problems with parameters (1234), (234), (124), (123), (23), (24), and (12) in the model. Solving a problem each time the associated node is encountered would result in a series of needless repetitive calculations. It is therefore important to construct and traverse a tree in a way that requires the least amount of effort and reduces redundant computations.

<figure 1 goes here>

The search procedure for selecting the best subset of all sizes is described here. The selection of the best subset of k, k+1,...,m parameters is a straightforward generalization of this procedure. The computer code developed by the authors does handle

the more general case.

In the implicit enumeration procedure, generally, not all the subproblems need be solved to optimality. For a current subproblem consisting of k parameters, if the objective value of the optimal solution to this subproblem is greater than the best objective value of previous subproblems of h parameters where $h < k$ the descendant nodes from the subproblem with k parameters will not yield improved solutions. Thus these nodes need not be examined.

Define z_k^u to be the upper bound on the objective value of an optimal solution with k parameters in the subproblem. Initially, every parameter is included in the model, namely, $J = \{1,2,...,m\}$ and the values of z_k^u, $k = 1.2,...,m$ are set to infinity.

The tree is inspected using a last-in-first-out (LIFO) branching rule. The subproblem chosen to be solved next is called the current candidate problem. When two descendants are created from the parent problem, the subproblem with a parameter removed from the parent problem becomes the current candidate problem. When no further progress can be made descending a branch, the algorithm backtracks up the tree and chooses the most recently created subproblem for inspection. Because of the LIFO branching rule, the current candidate problem is created with a minimal amount of effort and the tree can be described with parameter length arrays. Two arrays IPAR and ISTAT are utilized to define the current subproblem. The array IPAR is defined as follows:

$$IPAR(i) = \begin{cases} -k & \text{if the k-th parameter is forced out of the model at level i, i = 1, 2, ..., m;} \\ +k & \text{if the k-th parameter is required to be included in the model at level i, i = 1, 2, ..., m.} \end{cases}$$

The other array ISTAT has the following functions:

$$ISTAT(k) = \begin{cases} 0 & \text{when the k-th parameter is free;} \\ 1 & \text{when the k-th parameter is forced in the model;} \\ -1 & \text{when the k-th parameter is forced out of the model.} \end{cases}$$

At any stage of the algorithm, the partial assignment of subsets of parameters corresponds to a list of candidate problems. Once a candidate problem (CP) is selected, it is solved via a linear programming algorithm ([1] or [2]). The current solution at any stage is used to indicate a starting procedure for the next stage. A forward step consists of selecting a parameter based on (3) and fixing it out of the model. A backward step consists of requiring a free parameter to be included in the model. The complete tree has been inspected when all entries of the ISTAT array are positive.

A conceptual flowchart of the branch-and-bound procedure is illustrated in Figure 2.

<figure 2 goes here>

4. The Application of the Branch-and-Bound Algorithm Using a Dual Linear Programming Method

This section discusses how the dual linear programming method developed by Armstrong and Kung [2] is implemented within

the branch-and-bound algorithm described earlier. Two strategies employing the dual method are inspected. The first strategy is the use of a reoptimization start, and the second one is an implementation of a more powerful bounding test.

The dual of problem (2) is:

$$\text{(4)} \quad \text{Maximize} \quad \sum_{i=1}^{n} \pi_i y_i$$

$$\text{subject to} \quad \sum_{i=1}^{n} \pi_i x_{ij} = 0, \qquad j \in J$$

$$\pi_i \leq 1, \qquad i=1,2,\ldots,n,$$

$$\pi_i \geq -1, \qquad i=1,2,\ldots,n.$$

The two fundamental procedures to solve linear programming problems are the primal and dual algorithms. Because of the symmetry of linear programs (the dual of the dual is the primal), it is sometimes difficult to distinguish the two algorithms. The dual algorithm applied to (4) is the same as the primal algorithm applied to (2) and vice versa. A dual method will be termed to be an algorithm that maintains a feasible solution to (4) and strives to obtain a feasible solution to (2). A primal algorithm maintains a feasible solution to (2) and strives to obtain a feasible solution to (4). A detailed description of these two algorithms are found in [1] and [2].

At each stage of the dual algorithm, the values of the parameters are the simplex multipliers for (4). These multipliers can be calculated as $\beta^* = Y_B B^{-1}$ where β^* has dimension m(J), the cardinality of J, Y_B is a vector of dimension m(J) corresponding to the basic components of Y, where $Y = (y_1, y_2, \ldots, y_n)^T$, and B^{-1} is the current basis inverse of dimension m(J) by m(J).

4.1 Reoptimization Start

The optimal basic solution to a parent problem is stored and used as a start for the immediate descendant which has a parameter removed from the parent problem. Thus, the dimension of a basis for this descendant is one less than a basis for the parent problem. The process of initializing the basis and solution for the descendant is as follows. Let IB represent the index set of the basic variables and NB represent the index set of the nonbasic variables. Consider individually the constraints $\beta_j = 0$ for all β_j not forced in or out of the model by some previous restriction. Perform minimum ratio tests to determine, for each possible new restriction, the objective change during the first iteration and the basic π_s to leave the basis during this iteration. Choose β_r using (3). The observations associated with β_r are removed from the problem and π_s is removed from the basis creating a new basis of dimension one less than that of the immediate predecessor.

(5) $\pi_s = \{\pi_j$ removed from basis when constraint $\beta_r = 0$ is added$\}$

The new solution is dual feasible and the linear programming solution process can begin.

The variable leaving the basis will be set to the bound prescribed by the ratio test. The remaining nonbasic variables are set equal to their value in the optimal solution of the parent problem. The values for the basic variables are assigned to satisfy the constraint equations. This start enables the algorithm to determine an initial solution to the subproblem which should be a reasonable approximation of the optimal solution. The computational experience shows the efficiency of this start when

compared to an initialization procedure which does not utilize information obtained during the solution of subproblems considered previously.

4.2 Bounding Test

In addition to the bound check described in the general branch-and-bound scheme, an additional test to be performed during each phase 2 iteration is introduced. The purpose of this bounding test is to eliminate needless calculations when the best solution of the k-parameter subproblem cannot be improved.

In the dual method, basic feasible solutions are available at each iteration. For a current subproblem consisting of k parameters, if the objective value of a basic feasible solution, Z_k, is greater-than-or-equal to Z_k^u which is the objective value of the best k-parameter regression found thus far, the current subproblem need not be solved. This procedure evaluates a node (or a subproblem) without solving it to optimality. As described in [12], this bounding test is carried out prior to a simplex pivot. Thus, the effort of computations is reduced substantially.

5. The Application of the Branch-and-Bound Algorithm Using a Primal Simplex Method

Another linear programming method to evaluate the subproblems in the branch-and-bound procedure is a primal simplex approach to problem (2). Since this is a primal method, only the final basic solution is feasible for (4). Hence, the bounding test utilizing the basic feasible solutions in the dual method cannot be applied. Only the reoptimization start employing the

primal method will be described here.

The reoptimization process is very similar to the start procedure discussed in the dual approach. The main difference is the method to obtain the values of the nonbasic variables for the initial solution of the descendant problem.

If the basic variable, π_s, is selected to become nonbasic based on (5), the index s will be removed from IB and added to NB. The initial basis of full rank for the descendant problem, say $\hat{B}$, can be attained by means of the operation described earlier in Section 4.1 of this paper. However, the values of the nonbasic variables need to be computed to guarantee feasibility for (2). Their values are based on the sign of their reduced costs. The reduced costs of the nonbasic variables are given by:

$$(6) \quad \bar{y}_j = y_j - \beta^* X_j \qquad j \in NB$$

where X_j is the j-th row of X which is the observational matrix. The dimension of X is n by m(J).

From the reduced costs, the initial values of the nonbasic variables for the descendant problem are:

$$(7) \quad \pi_j = \text{sign } (\bar{y}_j), \qquad j \in NB$$

As indicated in Section 4.1 of this paper, the reoptimization start has its advantage in finding the initial solution to the descendant based on the results of the parent problem. The efficiency of this advanced start will be indicated in the computational tests reported in the next section.

6. Computational Experience

The branch-and-bound algorithms using the primal [1] and the dual [2] methods have been programmed in FORTRAN IV. All of the original information, including the observational matrix, is preserved by the program during execution. All the problems were solved on the CDC Dual Cyber 170/750 computer at The University of Texas at Austin Computation Center using an FTN compiler. The computer jobs were executed during periods when the machine load was approximately the same, and all solution times are exclusive of input and output. The total time spent solving the problem was recorded in central processor seconds by calling a Real Time Clock upon starting on the problem solution and again when the solution was obtained.

All the observations for the test problems have been drawn from various uniform and normal distributions using a random number generator. The tolerance value for zero was set at 1.0E-8. The number of parameters may not exceed 20 and the number of observations may not exceed 300. The user can easily extend these limitations by changing the dimensions on the appropriate working arrays in the program. The matrix of observations must have full column rank. The times are total execution time in CPU seconds and the number of iterations are updates of the basis inverse.

The branch-and-bound algorithm employing the dual linear programming approach with the reoptimization start was compared to an initialization procedure which does not utilize information obtained during the solution of subproblems considered previously. Thus, the dual algorithm will require a phase 1 procedure when the

reoptimization start is not used.Three sets of data consisting of 50 observations on 6, 8, and 10 parameters, respectively, were drawn from a random number generator. The computational results indicated that the reoptimization start enables problems to be solved approximately 10 to 20 times faster than the version of the algorithm without the advanced start procedure. Thus, all further comparisons are made with algorithms that include the advanced start.

The second phase of testing evaluated the implementation of the bounding test within the dual approach. The testing involved three codes for comparison purposes. The first code. is the primal version of the branch-and-bound algorithm with the feature of reoptimization start. The second code, TDUAL, includes the strategies of the reoptimization start and the bounding test in the dual version of the best subset algorithm. The third is a version of TDUAL without the option of the bounding test. Several different sizes of observations on 6, 8, and 10 parameters, respectively, were randomly drawn. The computational time and iteration count uniformly indicated that TDUAL is 5 to 10 times faster than the other two versions.

The final phase of computational testing involved comparing TDUAL with a code SUBSET written by Narula and Wellington and based on the algorithm of [13]. SUBSET does contain some options not available in the TDUAL code. These are the following:

1) Minimum sum of weighted absolute error and minumum sum of relative error are available as alternate criterion
2) A constant term may be required for all subsets.

TDUAL and SUBSET are written completely in standard FORTRAN. SUBSET implements a full tableau approach while TDUAL is based on a revised simplex method. Both codes utilize Gaussian elimination for update purposes.

A summary of the computational testing on TDUAL and SUBSET is given in Table 1. All the solutions obtained by the two codes matched out to seven significant places. No attempt was made to evaluate the stability of the codes. As can be seen, the advantages of the TDUAL code become more apparent as n increases in size. This can be attributed to rule (3) for choosing the parameter to restrict and the use of the dual algorithm with the advance start to solve the subproblems at each node.

All the results in Table 1 are for algorithms which guarantee the optimal regression from each subset. It is possible (see [10]) to obtain time-accuracy tradeoffs by considering near-optimal models. This type of modification is easy to implement in the computer programs tested here (a single line of FORTRAN code is changed). The fathoming is based on a function of the current incumbent other than the objective value. The optimal model is not guaranteed but savings in solution time can be significant. Table 2 gives the results of solving a set of test problems with a requirement that the regression be within 90%, 95% and 98% of the optimal. Even though only a certain percentage of optimality is guaranteed, the optimal solution was frequently obtained because of rule (3) to choose the parameter to remove from a subproblem. For example, when guaranteeing 95% of optimality, the optimal solution was obtained over 90% of the time.

7. Conclusion

In this paper a branch-and-bound algorithm to select the best subset of parameters in linear multivariate regression problems under the least absolute value criterion is presented. The algorithm is implemented with strategies involving the selection rule to restrict a particular parameter the fathoming test and the last-in-first-out (LIFO) branching rule for the inspection of the tree. Versions of the algorithm applying a dual as well as a primal simplex technique procedure were formulated and tested. A reoptimization start procedure is implemented in both the primal and dual version of the algorithm. In the dual version, the feature of a bounding test utilizing basic feasible solutions is also employed.

As indicated from the computational results, the advanced start procedure saves considerable computation time. With the addition of the bounding test in the dual approach, the dual version of the algorithm is consistently faster than the primal method, especially on problems where a large number of parameters are to be examined. In general, the branch-and-bound algorithm utilizing the dual approach with the advanced start and bounding test characteristics is most efficient for finding the best subset of regressors for least absolute value problems.

A computer code version of the algorithm is available from the authors for academic purposes.

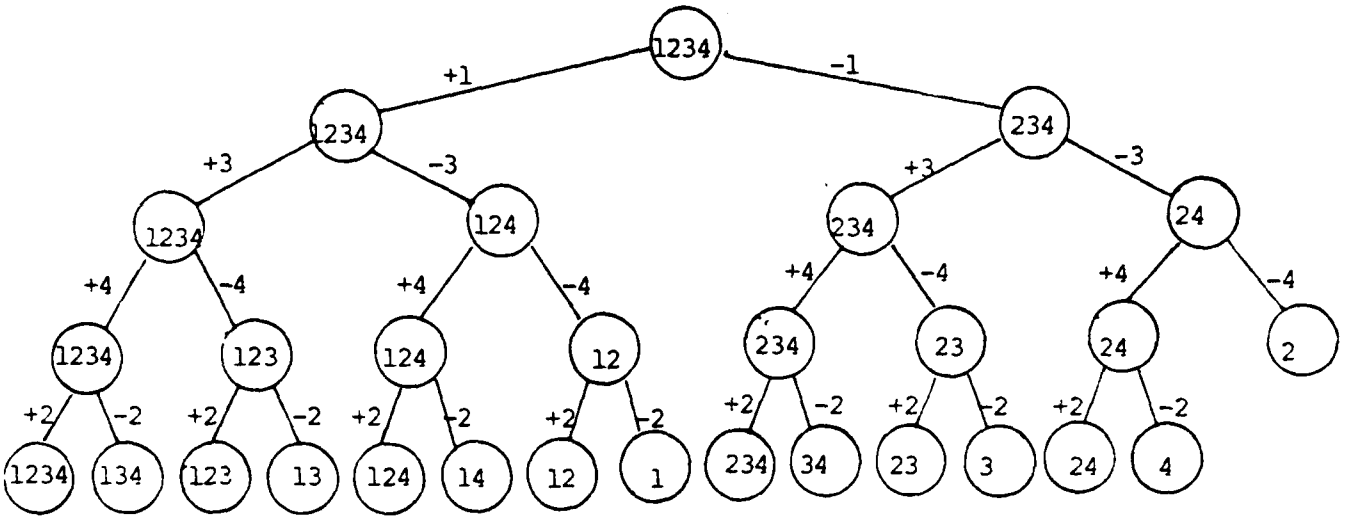
Acknowledgement: The authors wish to thank Professors S. C. Narula and J. F. Wellington for providing a copy of their best subset program for computational comparisons.

number of observations (n)	number of parameters (m)					
	m = 6		m = 8		m = 10	
	TDUAL	SUBSET	TDUAL	SUBSET	TDUAL	SUBSET
100	.193 (128)	.25 (270)	.781 (385)	1.12 (1351)	3.03 (1035)	5.39 (2965)
150	.381 (249)	.789 (486)	1.16 (659)	1.79 (839)	3.80 (1240)	7.37 (3081)
200	.564 (350)	1.16 (533)	1.35 (598)	3.04 (1124)	7.28 (3200)	17.83 (5735)
250	.855 (468)	2.11 (849)	2.61 (1293)	5.70 (1789)	8.70 (3460)	22.52 (5952)
300	1.33 (643)	2.85 (1019)	2.97 (1202)	6.23 (1703)	10.46 (3763)	22 20 (4906)

Table 1. Computational comparison of TDUAL and SUBSET obtaining the best subset for k = 1,2,....m. Three problems were solved in each combination of m and n. The upper entry in each row is the mean CPU time in seconds and the lower entry is the number of iterations.

		Percentage of Optimality Guaranteed			
		90	95	98	100
number of parameters	6	.495 (178)	.718 (318)	.828 (376)	1.33 (643)
	8	.834 (172)	.964 (242)	1.60 (590)	2.97 (1202)
	10	2.61 (250)	3.04 (512)	5.13 (1661)	10.46 (3763)
	12	11.11 (246)	11.43 (389)	12.94 (1238)	35.85 (7941)

Table 2: Computational comparison of TDUAL guaranteeing various percentages of optimality. The upper entry in each row is the mean CPU time in seconds and the lower entry is the number of iterations.



The numbers in each node indicate the indexes of the parameters of a subproblem. The negative number on the branch indicates the parameter to be taken out of the model, while the positive number on the branch states that the parameter is required to be in the model.

Figure 1: The Complete Tree For A Four Variable Problem

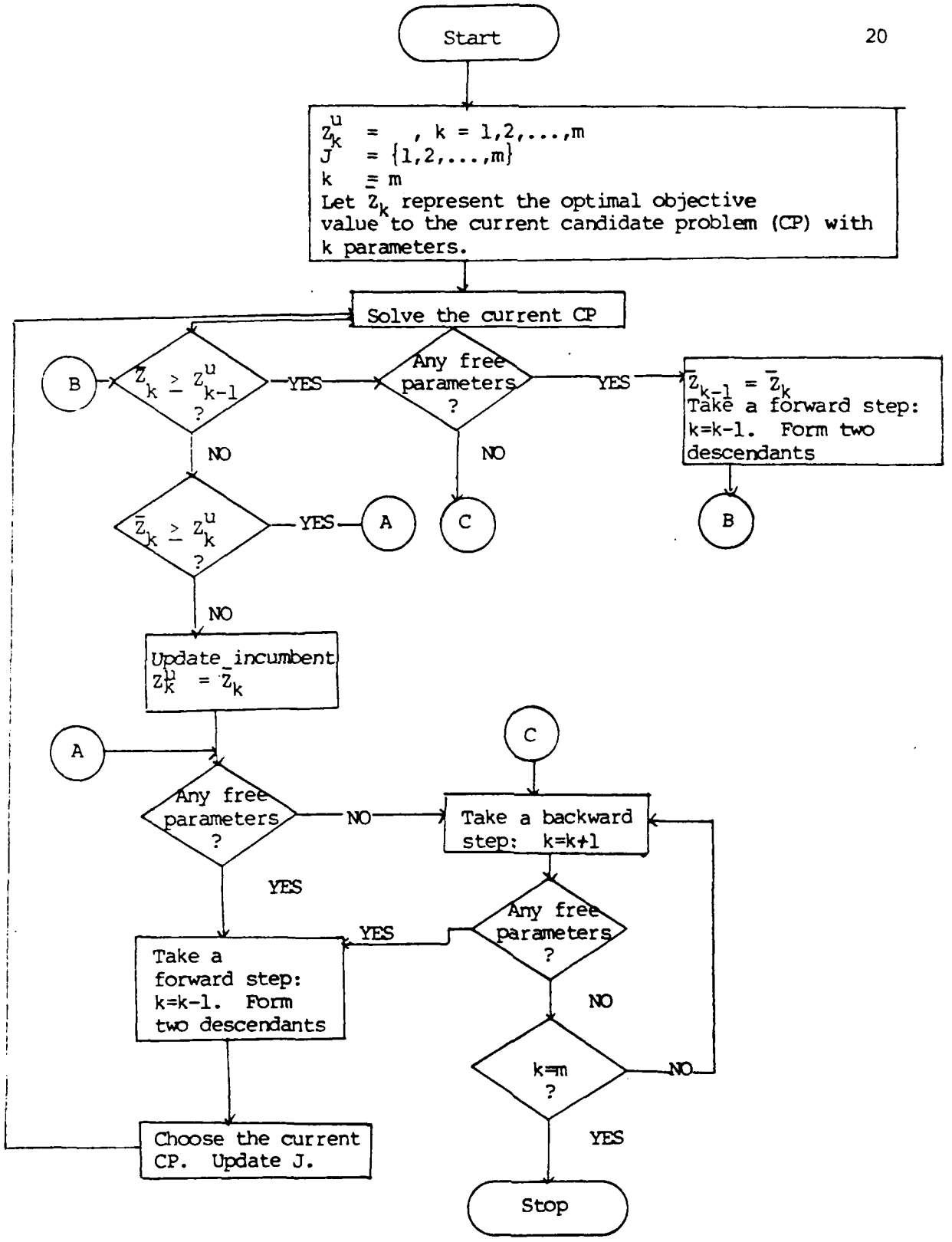


Figure 2: Flowchart for the branch-and-bound algorithm to obtain the best subset regression.

REFERENCES

1. Armstrong, R.D., Frome, E.L. and Kung, D.S., (1979), "A Revised Simplex Algorithm for the Absolute Deviation Curve-fitting Problem", Communications in Statistics, B8(2), 175-190.

2. Armstrong, R.D. and Kung, M.T., (1980), "A Dual Algorithm to Solve Linear Least Absolute Value Approximation", Research Report, CCS 370, Center for Cybernetic Studies, The University of Texas, Austin, Texas.

3. Beale, E.M.L., Kendall, M.G. and Mann, D.W. (1967), "The Discarding of Variables in Multivariate Analysis", Biometrica, 54, 357-366.

4. Charnes, A., Cooper, W.W. and Ferguson, R.O., (1955), "Optimal Estimation of Executive Compensation by Linear Programming", Management Science, 1, 138-151.

5. Charnes, A. and Cooper, W.W., (1961), Management Models and Industrial Applications of Linear Programming, Volumes I and II, New York, John Wiley & Sons, Inc.

6. Draper, N.R. and Smith, H., (1966), Applied Regression Analysis, New York, John Wiley & Sons, Inc.

7. Fama, E.F. and Roll, R., (1968), "Some Properties of Symmetric Stable Distributions", Journal of the American Statistical Association, 63, 817-836.

8. Furnival, G.M. and Wilson, R.W., (1974), "Regression by Leaps and Bounds", Technometrics, 16, 499-512.

9. Kennedy, W.J. and Gentle, J.E., (1980), Statistical Computing, New York, Marcel Dekker.

10. La Motte, L.R. and Hocking, R.R., (1970), "Computational Efficiency in the Selection of Regression Variables", Technometrics, 12, 83-93.

11. Martin, J., (1977), Computer Data-Base Organization, Englewood Cliffs, New Jersey, Prentice-Hall, Inc.

12. Meyer, J.R. and Glauber, R.R., (1964), "Investment Decisions, Economic Forecasting and Public Policy", Division of Research Memoir, Graduate School of Business Administration, Harvard University, Cambridge, Massachusetts.

13. Narula, S.C. and Wellington, J.F., (1979), "Selection of Variables in Linear Regression Using the Minimum Sum of Weighted Absolute Errors Criterion", Technometrics, 21, 299-306.

14. Rice, J.R. and White, J.S., (1964), "Norms for Smoothing and Estimation", SIAM Review, 6, 243-256.

15. Roodman, G., (1974), "A Procedure for Optimal Stepwise MSAE Regression Analysis", Operations Research, 22, 393-399.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1 REPORT NUMBER: CCS 396	2 GOVT ACCESSION NO.: AD-A100 460	3 RECIPIENT'S CATALOG NUMBER:
4 TITLE (and Subtitle): An Algorithm to Select the Best Subset for a Least Absoltue Value Regression Problem		5 TYPE OF REPORT & PERIOD COVERED:
		6 PERFORMING ORG. REPORT NUMBER:
7 AUTHOR(s): R. D. Armstrong and M. T. Kung		8 CONTRACT OR GRANT NUMBER(s): N00014-81-C-0236 ✓ N00014-75-C-25 9✓
9 PERFORMING ORGANIZATION NAME AND ADDRESS: Center for Cybernetic Studies, UT Austin Austin, Texas 78712		10 PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS:
11 CONTROLLING OFFICE NAME AND ADDRESS: Office of Naval Research (Code 434) Washington, DC		12 REPORT DATE: May 1981
		13 NUMBER OF PAGES: 24
14 MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15 SECURITY CLASS. (of this report): Unclassified
		15a DECLASSIFICATION/DOWNGRADING SCHEDULE:

16 DISTRIBUTION STATEMENT (of this Report)

This document has been approved for public release and sale; its distribution is unlimited.

17 DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18 SUPPLEMENTARY NOTES

19 KEY WORDS (Continue on reverse side if necessary and identify by block number)

Best Subset Regression Problem
Least Absolute Value
Linear Programming
Branch-and-Bound Algorithm

20 ABSTRACT (Continue on reverse side if necessary and identify by block number)

This paper considers the problem of obtaining the best subset of regressors under a least absolute value criterion. The model is the classic linear regression model with m explanatory variables and a dependent variable. The importance of the explanatory variables is measured by obtaining the minimum sum of absolute deviations when only k of the m explanatory variables are indluced in the model. An algorithm is presented to obtain the "best" subset of size k, k=1,...,m.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

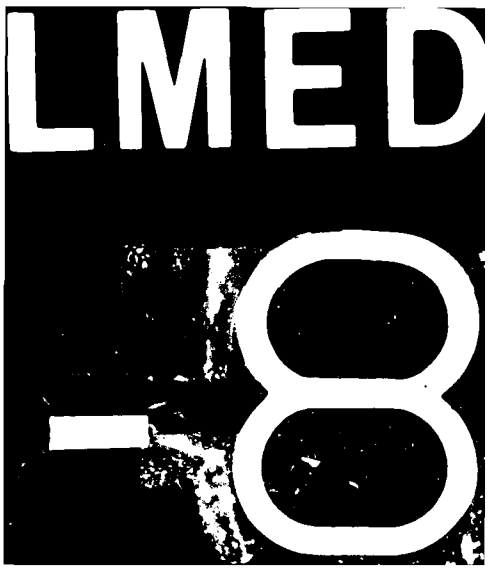
Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. Abstract (Cont'd)

Several algorithms to solve the best subset problem are available when the criterion for evaluation is least squares. However, recently statisticians have become increasingly aware of the limitations of least squares and have popularized "robust-resistant" estimation techniques. Least absolute values is such as technique. Special purpose computer codes which utilize the simplex algorithm of linear programming are used to solve the least absolute value regression problem.

This paper incorporates two of these specialized codes within a branch-and-bound algorithm to solve the best subset problem. The advantages and disadvantages of the two codes, one primal and one dual, will be discussed. Also, a detailed description of the branch-and-bound implementation and the results of computational testing will be given.


LMED
8